IN THE SPECIFICATION:

Please replace the paragraph beginning at Page 28, line 29 with the following:

In the present example, each FRU within a shelf is normally assigned an IP address dependent on the shelf in which it is installed. In order to achieve this, the CSSP 71 upon startup/reboot obtains the FRUID data from the midplane 171. This FRUID data is unique to the shelf (as the midplane is typically not field replaceable) and is usually stored in an EEPROM co-located with one of the LED indicator boards 183 or 184, which EEPROM is directly addressable by the service processor 74 via the midplane 171. Each FRU then uses an identifier including both the shelf-specific FRUID data and a FRU-specific ID for performing DHCP configuration. In the case of the CSSP 71, the switch 73 and the service processor 74 each perform DHCP configuration such that each obtains a different IP address. The identifier used by the switch 73 or service processor 74 for DHCP configuration may take the form: <shelf_ID; CSSP_ID; switch_or_serviceprocessor>.

Please replace the paragraph beginning at Page 30, line 1 with the following:

One management problem which can arise in the CSSP of the present example is where the major components (e.g. switch CPU and service processor CPU) are produced by different manufacturers. In this case there can be a conflict of data regarding the FRU manufacturing data such as production date, production time, and serial number. It is important for the data for both parts of the CSSP to be accessible, but it is also important that the CSSP as a whole is referenced with a single set of data for both devices' user interfaces.

Please replace the paragraph beginning at Page 30, line 9 with the following:

To avoid this difficulty, it is possible to configure the service processor 74 as master of the inventory and manufacturing data as it is normally the first point of access for system servicing. The service processor 74 can thus pass the correct data from (the FRUID) to the switch 73 upon initialisation. This data is then stored by the switch 73 in non-volatile memory (e.g. FLASH memory) in addition to its own hardware tracking and manufacturing data. Thus the switch 73 then provides the data provided by the service processor [[73]] 74 (obtained from

the FRUID) as its primary output, but is also able to output its original manufacturer's data if required.

Please replace the paragraph beginning at Page 38, line 32 with the following:

This arrangement also allows the management interface of the switch element 730 to be simplified. Taking the example of a command line interface (CLI), the service processor element CLI is operable to perform authentication and cryptographic operations (e.g., encryption/decryption operations) with an external management entity. However, as all management of the switch element 730 is performed via the service processor element 740, the switch CLI has no need for any of that functionality. Similarly, in the case of a GUI, the switch element GUI can work on information only (e.g. http content) and all authentication and encryption/decryption are provided by the service processor element 740.

Please replace the paragraph beginning at Page 40, line 1 with the following:

This system also allows more intuitive fault fixing. For example if a processing cartridge 43 hangs (i.e. experiences a software error which causes operations to halt but does not shut down the processing cartridge 43), the hang may occur such that erroneous data is repeatedly output via the network port to the switch element 73. The switch element 73 will notice this happening and will shut down that network port so as not to be affected by the erroneous data and will report the error to the fault management unit. The fault management unit will also be notified by the service processor element 740 that an error has occurred with that processing cartridge 43. The fault management unit will record the error messages and pass on details of the failure to the external management/service entity. These details may then be analysed to determine whether any reversement actions are required by the originator of the error message when a fault repair is attempted. Subsequently, an[[d]] attempt is made to restart the processing cartridge 43. An instruction is received from the external management/service entity to restart the processing cartridge 43. The fault management unit is consulted and as a result it is discovered that the switch element 730 has shut down its network port relating to that processing cartridge 43. Thus the service processor element 740 can increase the likelihood of the restart working by instructing the switch element 730 to re-open the relevant network port so that if the reboot of the processing cartridge 43 works, that processing cartridge 43 will be able to

communicate via the switch element. In the absence of such a system, it is possible that the reboot would be judged a failure regardless of actual success or failure as the data communication channel to and from the processing cartridge would remain closed.